Information Sciences 289 (2014) 25–40

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

How to adjust the distribution of nonzero elements in sparse representation: A granular locality-preserving approach

Yongchuan Tang *, Zejian Lee

College of Computer Science, Zhejiang University, Hangzhou 310027, PR China

ARTICLE INFO

Article history:
Received 21 January 2014
Received in revised form 1 July 2014
Accepted 27 July 2014
Available online 13 August 2014

Keywords:
Sparse representation
Face recognition
Digit recognition
Letter Recognition
Granularity
Locality-preserving

ABSTRACT

In this paper, we mainly discuss the importance of distribution of nonzero elements in sparse representation. In feature space of low dimension, limited number of nonzero elements are needed to represent the target and therefore the representation is naturally sparse. Ideally, if most of nonzero elements assemble around samples of the same class as the target, the reconstruction error tends to be small and the result is more likely to be correct. Therefore, it is necessary to introduce some discriminative information into the objective function to adjust distribution of nonzero elements. We propose the Granular Locality-preserving Classification (GLC) algorithms within fine, intermediate and coarse granularity, which incorporate distance metric, class labels and clustering results of *K*-means on training data as discriminative information. Experiments conducted on several benchmark data sets show that GLC algorithms are comparable with state-of-the-art classification methods.

1. Introduction

In recent years, sparse representation has been a popular research field in artificial intelligence, pattern recognition and related areas. This trend is inspired by studies in neuroscience which find that visual image is encoded by visual cortex of human brain with a sparse representation [30,22]. Different from the common meaning, in the rest of this paper, we refer to "sparse" when we mean it is primarily populated with zeros in certain matrix, vector, etc. Namely, sparsity refers to the fraction of zero elements in a matrix or vector [28].

Due to advance of $\ell 1$-norm minimization techniques, sparse representation, namely, to represent a target sample on basis sparsely, is widely studied and it has been proven to be a useful tool for many problems of pattern recognition, such as face recognition [33], visual tracking [20], phoneme classification [23], microaneurysm detection [39], traffic sign recognition [16] and many other problems in computer vision [32]. In [33], Wright et al. proposed the Sparse Representation-based Classification (SRC), which tries to find sparse solution of linear reconstruction of the target sample via the $\ell 1$-minimization:

$$\hat{\mu} = \arg\min_{\mu} \|\mu\|_1 \ s.t. \|X\mu - y\|_2 \leqslant \epsilon$$

where $y = [y_1, y_2, \ldots, y_m]^T$ is the query sample and $X \in R^{m \times n}$ is the training set itself. Sometimes the constraint is too strict to be feasible, so in practice the following Lasso criterion [43] is adopted:

* Corresponding author. Tel.: +86 0571 87953695; fax: +86 0571 87951250.
E-mail addresses: tyongchuan@gmail.com (Y. Tang), zejianlee@gmail.com (Z. Lee).

http://dx.doi.org/10.1016/j.ins.2014.07.035
0020-0255/

$$\hat{\mu} = \arg\min_{\mu} \lambda\|\mu\|_1 + \|X\mu - y\|_2^2$$

The objective function above is a combination of regularization term and reconstruction error term.

At last, SRC gives the judgement by examining reconstruction from which class giving the least error. Such framework shows good performance and boasts robustness to occlusion and corruption as well as insignificant choice of feature when it takes well-registered training samples as dictionary. At the same time, it still leaves a lot of space for further imagination and improvement.

Training data is not the only choice for coding basis. Based on low-rank matrix recovery, Ma et al. [18] develop a dictionary learning algorithm which yields a discriminative low-rank dictionary as the coding basis. Wang et al. [31] select codebooks which can be learned incrementally under locality-constraint as the basis. Liu et al. [16] use group sparse coding method allowing similar feature descriptors to share similar sparse representation on groups of codebooks. What is more, combining Nonnegative Matrix Factorization (NMF) [24], Liu et al. [17] choose the projections of the target spanned in each nonnegative subspace as the basis vectors. Zhu et al. [42] pick up several most contributive samples in the first round of reconstruction as the basis in the final round. Xu et al. [34] give a similar idea which select samples of several nearby classes in the first stage as candidates for the second stage. Yang et al. [37] devise an incremental dictionary learning method to find representative prototypes able to sparsely code samples from various classes. Zhang et al. [41] propose a hierarchical sparse coding algorithm to learn dictionary of basis blocklets for architecture style recognition. Based on the assumption that intra-class variations are sharable, Deng et al. [4] select training samples and intraclass variant dictionary together as coding basis. Later, they also propose a 'prototype plus variation' representation model for sparsity based face recognition [5]. Moreover, coding can be done on training samples of each class separately, which needs multiple times of minimization, as shown in [3,21,35]. In [25], Shafiee et al. discuss the role of dictionary learning and suggest that taking all training samples as basis gives the best performance but costs much more time for classification than dictionary learning methods do.

Sparse representation is characterized by the $\ell 1$-norm regularization term, but the fundamental problem about the effectiveness of sparse representation leaves open because of the lack of 'rigorous mathematical justification' [32]. Zhang et al. [40] argued that, rather than sparsity from $\ell 1$-norm, SRC is powerful because of its collaborative representation (the representation of training samples from all classes) but not the sparsity from $\ell 1$-norm. So they propose Collaborative Representation-based Classification (CRC_RLS) [40] with $\ell 2$-norm regularization term, which enables the minimization to have analytic resolution. Naseem et al. [21] also develop Linear Regression Classification (LRC) model representing an image as combination of class-specific galleries via least-squares method. Shi et al. [26] propose similar algorithm with QR factorization. In [7], Gao et al. use a Laplacian regularization term to preserve consistency of local feature. Besides regularization term, the error term also has alternative of $\ell 2$-norm. In [36], Yang et al. use MLE (maximum likelihood estimation) term to estimate the fidelity of reconstruction.

Furthermore, given the coding coefficient μ, other rules instead of Nearest Subspace [15] can be taken to identify the target y. Brown [3] introduces kernel trick to the residual term, and Zhang et al. [40] proposes the regularized residual term. In addition, Yang et al. [35] chooses the class which has the minimum of $\ell 1$-norm of the coefficient as the judgement, after coding the query on each class respectively, and such rule shows good efficiency.

Different from previous works mentioned above, we mainly discuss the distribution of nonzero elements in sparse representation in this paper. More in detail, we will show how the distribution of nonzero elements in the representation vector influences efficacy of algorithm, and give our solution to adjust the distribution. What is more, we propose another simple but effective rule to judge the target based on sparse coefficients.

It is argued in [26] that the assumed linear dependence among homo-class samples in [33] does not exist, and in high dimension space $\ell 1$-minimization still gives dense solution. However, in space of low dimension, only small amount of independent samples are needed to represent the target, thus sparse representation is still meaningful. In sparse representation, since only small amount of elements are nonzero, distribution of these nonzero elements plays a critical role in the classification. Ideally, if most of the nonzero elements are just corresponding to training samples of the same class as the target, i.e., the target is mainly represented by the homo-class samples, reconstruction error of this correct class is smaller and therefore we are more likely to have correct classification. Such homo-class representation is the basic pursuit of sparse representation. Thus, it is necessary to incorporate some discriminative information into the objective function so as to adjust the distribution mentioned above. But what kind of information to choose and how to incorporate the information are still open challenges.

We give our solution by designing new objective functions which introduce three kinds of granular information: metric of distances between training samples and the target, the known class labels of training set, and clustering results of K-means on training data. The reason why we take metric of distances into account is because of the assumption that samples in smaller distances are prone to be of the same class. So we hope the majority of nonzero elements of the solution are just the coefficients of training samples near the test target. To put it more explicitly, the closer the base element is to the target sample, the more probably the corresponding coefficient will be nonzero, or even the larger the value of that will be; at the same time, the remoter the base element is to the target sample, the more likely the coefficient will be zero. Distinct from previous works [31,3], we do not use selective method like K-nearest-neighbour (KNN) to enforce closeness. Instead, our algorithm is able to automatically yield locality-preserving coefficients. Furthermore, since we expect homo-class representation, we should consider the labels of training data via dividing and treating them differently by class. But label information is

class-wise, while distance information is sample-wise, so we come up with a solution of compromise in intermediate granularity which divides the whole training data to multiple groups according to intersection of labels and clusters given by *K*-means.

After the minimization of objective functions, we gain the representation coefficients, and use another rule instead of Nearest Subspace [15] to judge the target. To take advantage of the adjusted distribution of nonzero elements, we just identify the target as the class whose coefficients has the maximal $\ell 1$-norm. The $\ell 1$-norm of coefficients of each class directly embodies how much the nonzero elements gather around samples of this class and how this class contributes to the representation of the target. Such rule gives a more direct insight of distribution of elements than Nearest Subspace [15] does. That is why we adopt the maximal $\ell 1$ rule.

Experiments are conducted on several benchmark databases for different recognition problems to show that our objective function does adjust the distribution of nonzero elements, and that our algorithm can have comparable or sometimes better performance than other state-of-the-art methods.

The rest of the paper is organized as follows. In Section 2, we give a brief review of Sparse Representation-based Classification (SRC). Section 3 discusses the causes of sparsity and the importance of distribution of nonzero elements in sparse representation, while Section 4 introduces our algorithm. Experimental results will be presented in Section 5. Finally, we give discussion on several issues in Section 6 and conclude our paper in Section 7.

2. A brief review of SRC

Sparse Representation-based Classification (SRC) [33] is a classification framework for object recognition. The pursuit of sparsity is motivated by the simple observation that only the training samples of the same class often suffice to represent the target test sample. Given that training set is of larger number of classes, the representation from the training samples of all the class is naturally sparse. So SRC takes the following assumption.

Assumption 1 (*Over-completeness Assumption*). Sufficient training samples are available from each class so that any test sample can be represented as a linear combination of the training samples from the same class.

Based on such assumption, SRC tries to find the linear reconstruction of the target sample on an over-complete dictionary, which is training set itself, and the solution of reconstruction is expected to be sparse. In fact, it is the sufficient representation from the homo-class samples that leads to sparsity, and now SRC pursues the sparsity of representation so as to search the effective reconstruction from the same class backwards. What is more, as long as the structure of data supports it, sparse representation can be applied in under-complete data sets.

Intuitively, we can seek such sparse solution by minimizing the $\ell 0$-norm of the coefficient, which directly counts the number of nonzero elements in the vector. However, the problem turns out to be NP-hard [1] but has equal solution to $\ell 1$-minimization when this solution is sparse enough [6]. Therefore, such reconstruction process above can be computed via $\ell 1$-minimization. Based on the sparse coefficients, the Nearest Subspace [15] rule is usually adopted to identify the label of the sample.

Given a labeled training set $X \in R^{m\times n}$ of k distinct classes, each column vector is a training sample. In addition we define $X_i \in R^{m\times n_i}$ as training samples of the ith class and therefore $X = [X_1, X_2, \ldots, X_k]$. To classify the new sample $y \in R^m$, SRC aims to find the solution of (1):

$$\hat{\mu} = \arg\min_{\mu} \lambda\|\mu\|_1 + \|X\mu - y\|_2^2 \tag{1}$$

where $\|\cdot\|_1$ means the $\ell 1$-norm. Then SRC identifies y as

$$identity(y) = \arg\min_i r_i(y) \tag{2}$$

where residuals $r_i(y)$ is computed by

$$r_i(y) = \|X\delta_i(\hat{\mu}) - y\|_2. \tag{3}$$

$\delta_i(\hat{\mu}) \in R^n$ is a new vector whose nonzero elements are only those in $\hat{\mu}$ associated with the ith class. The procedure of SRC is summarized as follows.

Algorithm 1. The SRC Algorithm

1. Normalize the columns of X to have unit $\ell 2$-norm.
2. Solve the $\ell 1$-minimization problem:

$$\hat{\mu} = \arg\min_{\mu} \lambda\|\mu\|_1 + \|X\mu - y\|_2^2$$

3. Compute the residual $r_i(y) = \|X\delta_i(\hat{\mu}) - y\|_2$ for $i = 1, \ldots, k$
4. Output: $identity(y) = \arg\min_i r_i(y)$

The sparse representation boasts insignificant choice of feature and robustness to occlusion and corruption when it is conducted on well-registered samples and it takes training samples themselves as dictionary. To put it more clearly, when sparse representation takes training set as basis, the choice of features will be much less critical as long as the dimension of features is sufficiently large, and errors due to occlusion and corruption, which is believed to be sparse compared to standard basis, can be handled uniformly under this framework.

3. Importance of distribution of nonzero elements

3.1. Causes of sparsity

The essential idea of sparse representation is to search a homo-class representation with constraint of sparsity. The hypothesis behind Assumption 1 of SRC is the linear dependence among samples of the same class, which is thought to be the cause of sparsity and which is taken advantage of to search the homo-class representation. However, it is reported in [26] that there is not such simple linear dependence even in typically used data set. In [26], Shi et al. analyze the singular values of the well-known AR database, and point out that all singular values are relatively large, suggesting that the expected linear dependence does not exist intrinsically in homo-class data, and that solution of sparse coding on such data is inevitably dense.

We also conduct similar experiments on Extended Yale Face database B [8,14]. Detail of this data set is given in Section 5.1. We simply flatten the original pictures in the database to column vectors and gain a matrix of $32{,}256 \times 2441$. And then we calculate the log of singular values of this matrix and plot them in descendent order in Fig. 1. If Assumption 1 holds and each sample can be represented as a nearly linear combination of homo-class data, rank of this matrix will be lower than the total number of samples and the certain amount of smallest log of singular values will drop to very low negative ones. However, the log of singular values do not show expected characteristic but all lie above zero (more precisely, above two), suggesting the linear independence among each sample. For comparison, we also generate a synthetic data in which samples of different classes are spanned on different independent linear subspaces. The singular values of synthetic data show the expected phenomenon supporting Assumption 1, but are totally different from those of Yale B. As a result, we argue that there may be no intrinsic linear dependence among samples of the same class. Or, such linear dependence is seriously violated by noise of high variance so that it is hard to recover. In such space of high dimension, the matrix of data is likely to be of high rank, which is close or even equal to number of sample. Any new target must be linear approximated by the whole training set and thus the representation is inevitably dense.

However, sparse representation is still meaningful in feature space of low dimension. Generally, after preprocess such as feature extraction and dimension reduction, data is transformed into very compact representation in feature space of low dimension m, and $m \ll n$, the number of data entries. Under such circumstance, the rank of data matrix is also reduced to m or below. Therefore, a new vector can be linearly represented by any m independent vectors in training set, while the representation coefficient is naturally sparse and the measure of sparsity is $\frac{n-m}{n}$, nearly one. Such sparsity is caused by the low rank of feature rather than by linear dependence described by Assumption 1. In the rest of paper, we do not assume that Assumption 1 holds and just discuss the sparsity caused by low rank in space of low dimension.

Pursuit of sparsity only is not enough to ensure a meaningful homo-class representation. Since any m independent vectors suffice to represent a new sample accurately, it is not guaranteed that the m vectors are of same class as the target, and that the sparse coefficient can give meaningful discriminative information leading to correct classification. Ideally, if the target is mainly represented by m homo-class samples, the following Nearest Subspace rule tends to give correct judgement. Wright et al. [33] argue that $\ell 1$-minimization may recover highly neighbourly polytope of target and therefore the correct sparse representation, but such recovery is not ensured. It is still very important to further bias the selection of these m vectors, and to enforce explicit constraints such that these m vectors are mainly the samples of same class as the target. This is equivalent to that of adjusting the distribution of nonzero elements, simply because these m vectors are corresponding to nonzero entries in μ.

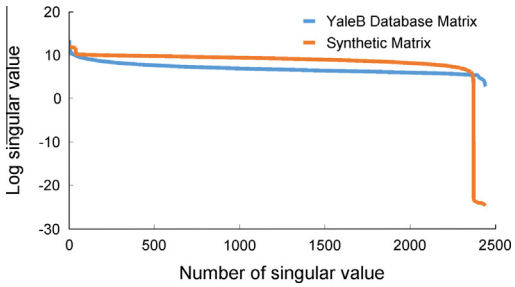

Fig. 1. Log of singular values of matrix generated from Extended Yale Face database B.

3.2. Distribution of nonzero elements

Before discussing how to adjust distribution of nonzero elements, we first extend the objective function in (1) as follow.

$$\hat{\mu} = \arg\min_{\mu} \sum_{j=1}^{n} \lambda_j |\mu_j| + \|X\mu - y\|_2^2 \tag{4}$$

This target function degenerates into (1) when $\lambda_j = \lambda$ for all $j = 1, \ldots, n$. This is the general form of Lasso problem and can be reformulated as the standard one. λ_j for $j = 1, \ldots, n$ are regularization parameters of different entries in μ. Although not necessarily, μ_j tends to be zero as λ_j increases [13]. By manipulating each λ_j and enforcing different impact on μ_j, it is possible to adjust the distribution of nonzero elements in μ.

We let $\lambda_j = \lambda\alpha_j$ and rewrite (4) as follows.

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{j=1}^{n} \alpha_j |\mu_j| + \|X\mu - y\|_2^2 \tag{5}$$

Like (1), the λ still control the influence of regularization term, while at the same time α_j gives different penalty impact on μ_j. Given the same λ in this minimization, the larger α_j is, the heavier the penalty impact is and thus the more probably μ_j is zero; conversely small α_j is apt to result in nonzero μ_j. Ideally, nonzero elements will gather around samples of same label, if we assign small value to α_j when μ_j is related to homo-class samples. As a result, such distribution can yield more accurate prediction.

We will support this argument by conducting and analyzing three ideal experiments. In Experiment A and C, we replace the objective function of SRC with (5) and assign the value of α according to the label of the test samples, and in Experiment B we just perform SRC. We first do these experiments on CMU Pose Illumination and Expression (PIE) database [27], detailed information of which will be described in Section 5.1, and we just take the PCA feature whose dimension is 30. The regularization parameter λ is set as 0.01, and the only difference is the value of α.

Assignment of α in different experiments:

- Experiment A: α_j is set to 2 if x_j is of the different class from y, otherwise α_j is 1.
- Experiment B: All α_j is 1, namely, it degenerates to the objective function of SRC.
- Experiment C: α_j is set to 0.5 if x_j is of the different class from y, otherwise α_j is 1.

The classification results are 99.78%, 95.86% and 12.41% respectively. We then evaluate the distribution of nonzero elements via the following measurement. Given that y belongs to Class i, the formulated definition is shown as follows.

Definition 1 (*Proportion of ℓ1-norm of Coefficients associated with Homo-class Samples*).

$$P(\mu) = \frac{\|\delta_i(\mu)\|_1}{\|\mu\|_1} \tag{6}$$

This measurement is the proportion between ℓ1-norm of homo-class coefficients and μ, and it examines how much the magnitude of homo-class coefficients takes up that of μ. This directly shows whether the distribution of nonzero elements gather on the homo-class samples. We refer to it as Homo-class Proportion for short in the rest of paper.

In Experiment A, the Homo-class Proportion is 76.1%, suggesting that homo-class coefficients takes up a large proportion in μ, while in the rest two the Homo-class Proportion is only 59.41% and 11.45% respectively.

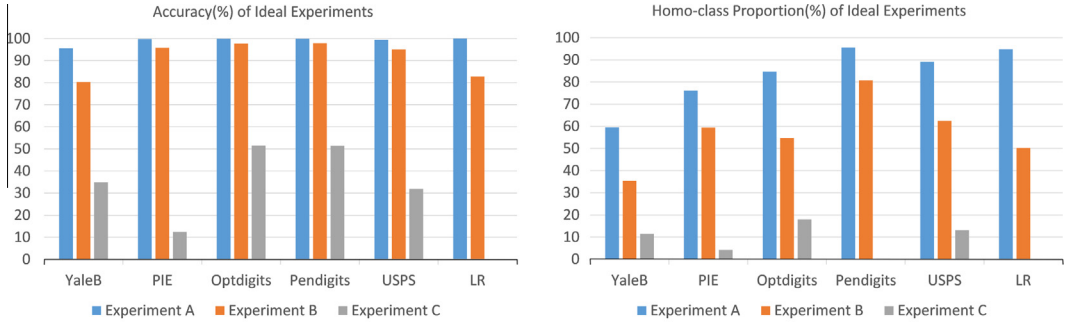

Fig. 2. (a) Accuracy of ideal experiments; (b) Homo-class Proportion of ideal experiments.

Recognition accuracy and Homo-class Proportion of all databases mentioned in Section 5.1 within the same ideal experiments are given in Fig. 2a and b. Generally, different assignments of α play critical roles in the distribution of nonzero elements. In most of the time, higher Homo-class Proportion leads to more accurate classification. All these results exemplify the influence of α on distribution of reconstruction coefficients and consequently the accuracy of classification. One approach to adjust the distribution of nonzero elements in sparse representation is to influence the magnitude of coefficients via controlling regularization parameters in general form of Lasso problems. So solution to distribution adjustment now turns out to be proper assignment of α. However, in practice we can hardly take the rule above, simply because we just do not know the label of the target. So what kind of information can we use to set α appropriately? What policy can we take? We will give our solution in the next section.

4. Granular locality-preserving classification

4.1. Locality-preserving reconstruction in fine granularity

To cope with the problem described in the previous section, we first give the following assumption:

Assumption 2 (*Closeness Assumption*). The test sample is inclined to be of the same class as training samples close to it in feature space; and it is likely of different category from samples having long distances off.

The distance in feature space measures dissimilarity of samples, and it is natural to judge two samples close to each other as the same label. Based on this assumption, we choose distance between x_j and y as α_j, and propose an objective function into which we introduces closeness metric as discriminative information. We denote d_j as the distance between the jth training sample x_j and the test sample y for $j = 1, 2, \ldots, n$. Such distance can be Euclidean distance or other. We choose Euclidean distance at default in our algorithm. Other types of distance will be discussed in Section 6.1.

We code y over X via the following $\ell 1$-minimization.

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{j=1}^{n} d_j |\mu_j| + \|X\mu - y\|_2^2 \tag{7}$$

The regularization term in (7) plays two roles. First, it introduces sparsity into the solution since it is $\ell 1$-norm. Second, due to the regularization term, elements of μ corresponding to training samples near the target y will be nonzero or even have larger values, while those related to the remote ones will probably be zero. Suppose x_j is the closest training sample to y, and thus it has the smallest d_j. As a result, μ_j is likely to be nonzero or even the largest element in μ. Conversely, if x_j is the most remote training sample from y, μ_j is expected to be zero.

The solution of (7) is not only sparse but also characterized by certain closeness. We show these characteristic via a simple illustration over a toy data set in Fig. 3a and b. We try to code the point shown as "$*$" over another nine points shown as "O" via minimization of (1) and (7) respectively. Points whose reconstruction coefficients are over 0.01 are emphasized in red and bold style. Fig. 3a shows solution of (7) enables the "$*$" to be represented by the nearest two points, while in Fig. 3b solution of (1) is not that close.

To sum up, (7) takes the distance measure as discriminative information and allows the distribution of nonzero elements to preserve closeness. We must point out that the objective function (7) coincides with the form of LCC [38] when LCC takes the whole training set as anchor points and $\ell 2$-norm as localization term. Nevertheless, the goal of LCC is to approximate the

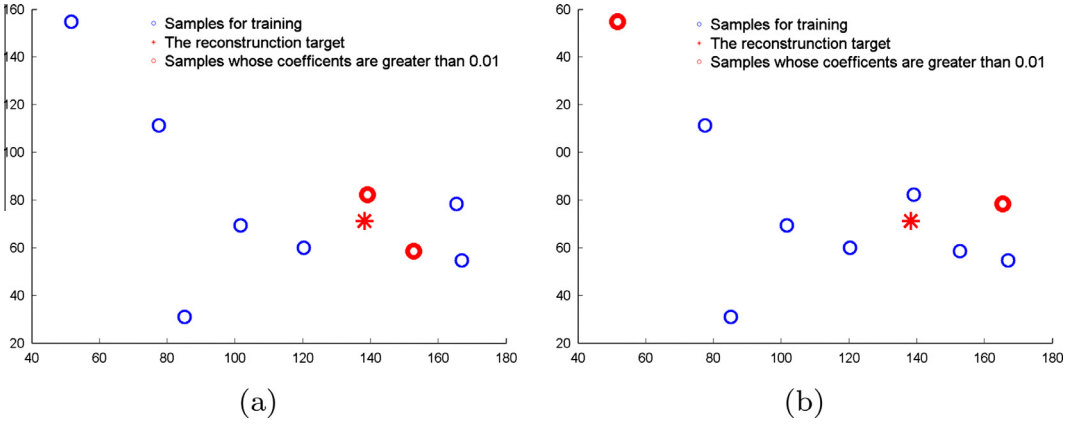

Fig. 3. (a) Reconstruction on toy data via (7); (b) reconstruction on toy data via (1) of SRC.

local linear relation on nonlinear manifold and to give a local coordinate coding of the target, while (7) aims to search a sparse and homo-class linear representation of the target. The motivation behind is different.

4.2. Locality-preserving reconstruction in coarse and intermediate granularity

Choosing d_j as α_j just takes the metric of distance into account, but ignores information of original class labels, which is also significant. Since we expect homo-class representation, we should take the class labels into consideration. Combined with the way of assignment in former idealized experiments, we can take distance from the target to each class as α_j. The distance between y and the nearest sample of each class is chosen at default. If training sample x_j belongs to ith class, and $\tilde{d}_i$ is

$$\tilde{d}_i = \min\{d_j | x_j \in X_i\}$$

every α_j is assigned as corresponding $\tilde{d}_i$. The minimization is changed as:

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{i=1}^{k} \tilde{d}_i \sum_{x_j \in X_i} |\mu_j| + \|X\mu - y\|_2^2 \tag{8}$$

This objective function further exploits the label information.

Let's further compare (7) and (8). (7) assigns each α_j respectively, and (8) gives a solitary value to all α_j of the same class. So the granularity of former one looks too fine, while that of latter one seems too coarse. Compromise of these two means may be found by further utilizing more information, such as distribution of the training data. We design the intermediate granularity version by grouping training data according to their labels and which clusters they belong to in result of K-means.

Why should result of K-means be taken into consideration? Unsupervised learning methods such as K-means generally capture the distribution of data and enforce mutual similarity within each cluster. Although the labels of training data are given, combining "advice" from K-means is also beneficial. K-means can retain certain locality in each clusters and divide the training set into groups of convex shape. This helps us to assign proper value to α_j of each entry within intermediate granularity. Detail of this rule to assign α_j is given as follows. Firstly, we conduct K-means on the training data and gain K clusters $[C_1, C_2, \ldots, C_K]$. Secondly, for the lth cluster C_l where $l = 1, 2, \ldots, K$, we divide C_l by class label of each entry, then group the homo-class samples together, and have c_l groups in the lth cluster if there are only samples of c_l different classes. After that, we have c groups for $c = \sum_{l=1}^{K} c_l$ and $c \leqslant k \times K$. Finally, for $p = 1, 2, \ldots, c$, all α_j for $x_j \in group_p$ are assigned as $\underline{d}_p$, which is defined as

$$\underline{d}_p = \min\{d_j | x_j \in group_p\}$$

The formulated minimization in intermediate-granularity version is shown as follows.

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{p=1}^{c} \underline{d}_p \sum_{x_j \in group_p} |\mu_j| + \|X\mu - y\|_2^2 \tag{9}$$

In addition, (9) reduces to coarse-granularity version (8) when $K = 1$, and to fine-granularity version (7) when $K = n$, the number of training samples. The precise selection of K is a challenging problem, and we recommend that K is set as the number of labels at default so as to divide the data properly.

To illustrate the three ways of assignment to α discussed above more clearly, we give more vivid demonstrations in two groups of pictures. The first group of pictures is shown in Fig. 4. Different colours mean different classes. Squares in lighter colours means smaller distances from the training samples to the target, and vice versa. Policy for assignment of α in (7) is depicted in Fig. 4a, (8) in Fig. 4b, and (9) in Fig. 4c. In Fig. 5, we pick up 300 training samples of Optdigits [2] and the second testing sample, and then project them into 2 dimension space with LPP [11] to give a more intuitive demonstration. The red point of diamond shape is the target. The shapes of points indicates their labels. The colour is consistent with the shape in Fig. 5a and b, while in Fig. 5c it shows the clustering result of K-means. Dashed ovals emphasize groups of data and arrows symbolize the assignment of distances.

In practice, Euclidean distance varies a lot from data sets, so we normalized them to $[1, +\infty)$ by dividing the minimum of them. Minimization of our three objective functions (7)–(9) are convex problems which can be solved via CVX [9,10]. Due to the complexity of $\ell 1$-norm regularization, such minimization often takes up 99% of computational time in our algorithms and may make the computation intractable for large data sets. To solve large-scale optimization within our objective functions, another toolbox SPAMS [19] is recommended. These are the reconstructions based on various rule to set α. Ideally, nonzero elements in μ assemble around samples nearby, and distribution of nonzero elements is skewed toward the vicinity of the target.

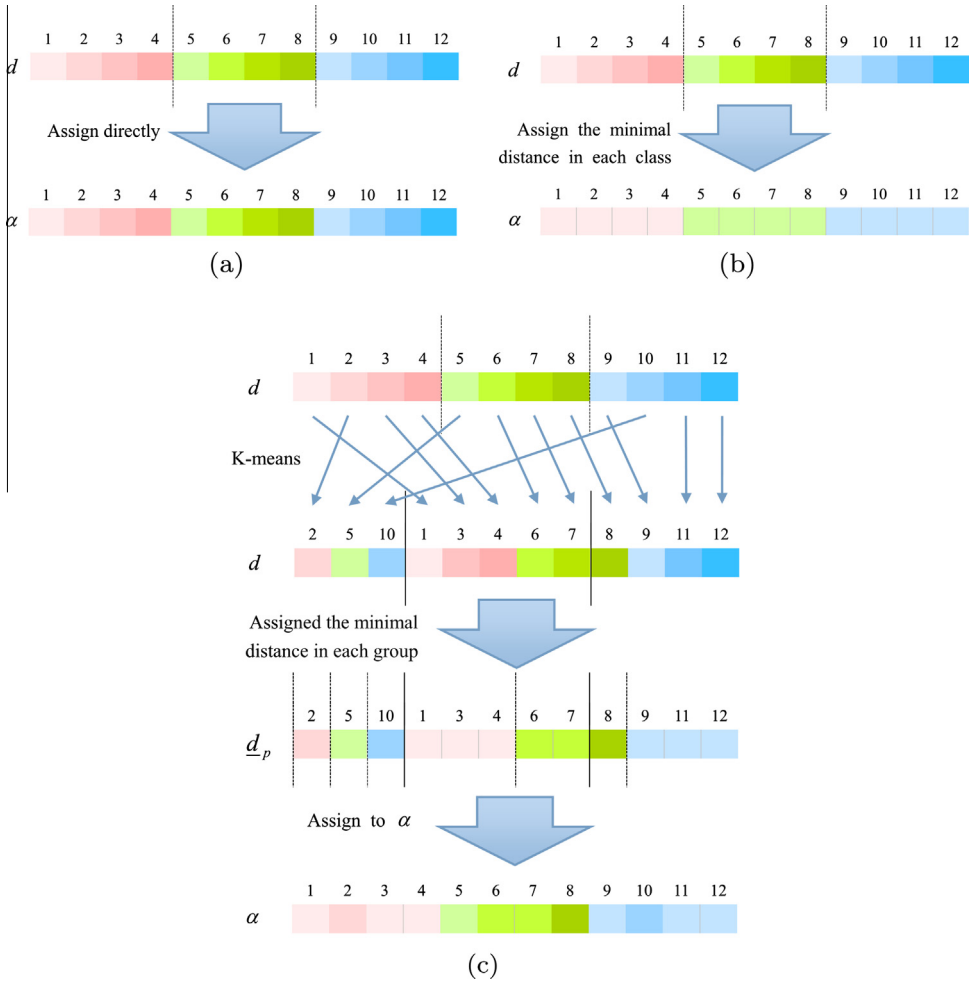

Fig. 4. (a) Assignment of α in fine granularity; (b) assignment of α in coarse granularity; (c) assignment of α in intermediate granularity.

4.3. Maximal $\ell 1$-norm of coefficients

With the reconstruction coefficient μ, instead of Nearest Subspace [15], we adopt another rule to make judgement, which is to judge y as the class coefficients of which gives the maximal $\ell 1$-norm.

This rule simply checks $\|\delta_i(\mu)\|_1$ of the ith class for $i = 1, \ldots, k$ and identifies the target as the class having the maximal one. The $\ell 1$-norm of $\delta_i(\mu)$ is a direct reflection of whether the distribution of nonzero elements bias towards the ith class and how samples of the ith class contribute to the reconstruction, and such rule gives more intuitive insight of distribution of nonzero elements than Nearest Subspace, which compares distances between the target and projections on different subspaces. We refer to this rule as maximal $\ell 1$ in the rest of paper.

As is mentioned in Section 1, the previous work Cℓ1C [35] takes the minimal $\ell 1$-norm of coefficients from each class as judgement. Why do we choose the largest one while Cℓ1C chooses the opposite? Cℓ1C regards $\ell 1$-norm of coefficients as reconstruction weights and takes a prior that homo-class representation will give rise to minimal reconstruction weights, so Cℓ1C codes the query on samples of each class respectively and identifies the query as the class which give the minimal one. On the other hand, our proposed algorithms take the whole training samples to represent the target, and view $\ell 1$-norm as the measurement of distribution of elements. As a result, to take advantage of the adjusted distribution of nonzero elements, GLC algorithms choose the class giving the largest $\ell 1$-norm of coefficients.

The formulated maximal $\ell 1$ rule is

$$identity(y) = \arg\max_i \|\delta_i(\mu)\|_1$$

Finally, procedures of our proposed Granular Locality-preserving Classification (GLC) in three kinds of granularity are summarized as follows. In the rest of the paper, we refer to them as GLC-fine, GLC-coarse and GLC-intermediate respectively.

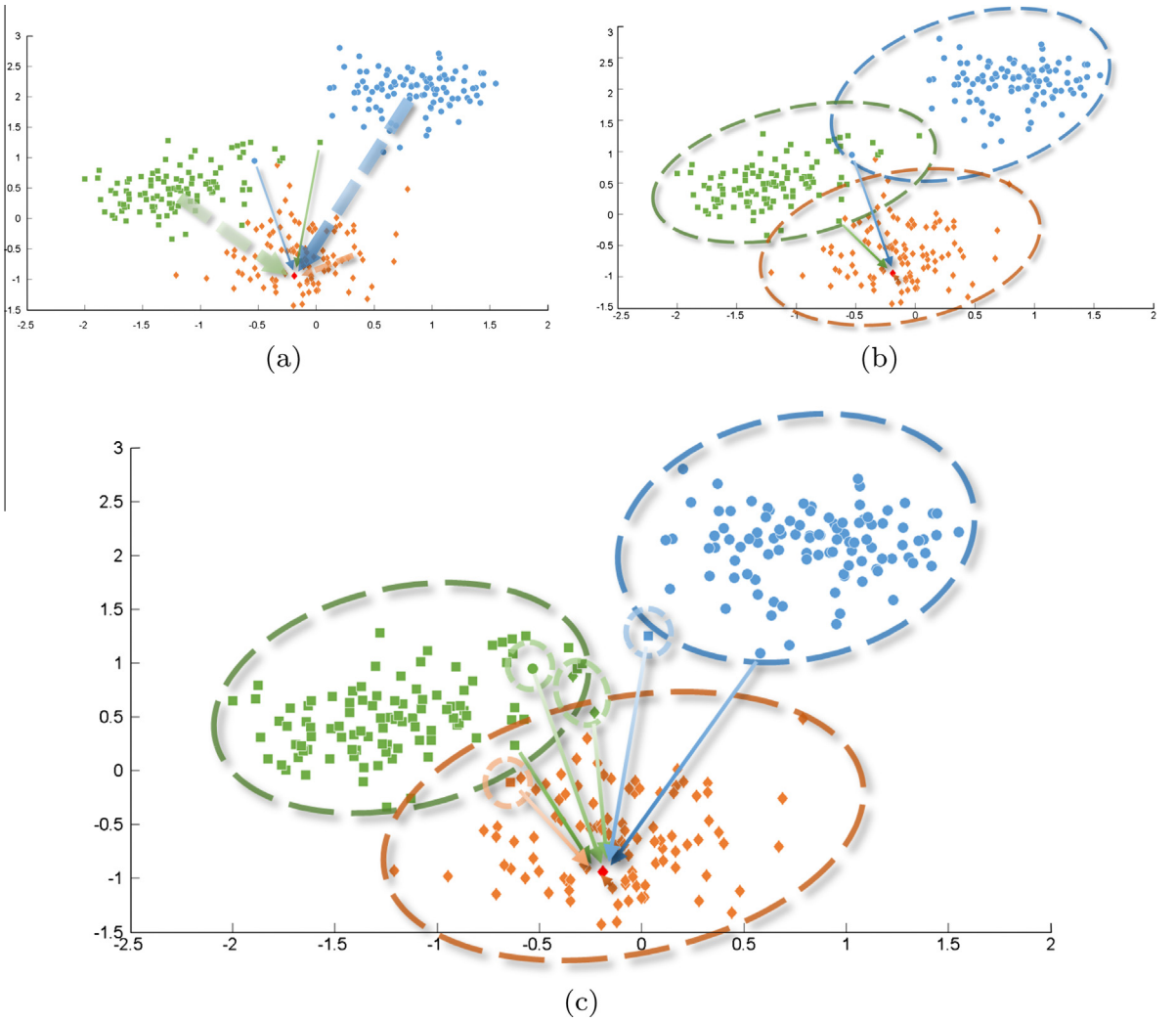

Fig. 5. (a) Assignment of α in fine granularity. α_j is set as d_j for each sample. For simplicity we just draw thick translucent arrows for each label and slim arrows for two outliners. (b) Assignment of α in coarse granularity. α_j is set as $\tilde{d}_i,\ x_j \in X_i$ for each sample, shown as arrows of different colours. (c) Assignment of α in intermediate granularity. Dashed circles point out different $group_p$. α_j is set as $\underline{d}_p,\ x_j \in group_p$ for each sample, shown as arrows of different colours. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Algorithm 2. GLC Algorithm in Fine Granularity (GLC-fine)

1. Normalize the columns of X to have unit $\ell 2$-norm.
2. Solve the $\ell 1$-minimization problem:

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{j=1}^{n} d_j |\mu_j| + \|X\mu - y\|_2^2$$

3. Output: $identity(y) = \arg\max_i \|\delta_i(\mu)\|_1$

Algorithm 3. GLC Algorithm in Coarse Granularity (GLC-coarse)

1. Normalize the columns of X to have unit $\ell 2$-norm.
2. Solve the $\ell 1$-minimization problem:

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{i=1}^{k} \tilde{d}_i \sum_{x_j \in X_i} |\mu_j| + \|X\mu - y\|_2^2$$

3. Output: $identity(y) = \arg\max_i \|\delta_i(\mu)\|_1$

Algorithm 4. GLC Algorithm in Intermediate Granularity (GLC-intermediate)

1. Normalize the columns of X to have unit $\ell 2$-norm.
2. Conduct K-means on X and divide all clusters into c groups according to class labels.
3. Solve the $\ell 1$-minimization problem:

$$\hat{\mu} = \arg\min_{\mu} \lambda \sum_{p=1}^{c} d_p \sum_{x_j \in group_p} |\mu_j| + \|X\mu - y\|_2^2$$

4. Output: $identity(y) = \arg\max_i \ \|\delta_i(\mu)\|_1$

5. Experiments

Our experiments are performed on several benchmark data sets. Besides data for face classification, digit recognition and letters recognition data are also used, including Extended Yale Face Database B [8,14], CMU Pose Illumination Expression (PIE) Database [27], Optdigits [2], Pendigits [2], USPS [12], Letter Recognition database [2]. We will show GLC algorithms can have comparable or sometimes better performance than other state-of-the-art methods. What is more, we will examine the distributions of nonzero elements among different methods.

5.1. Introduction of data sets

Details of the databases we use in experiments are briefly described as follows.

Extended Yale Face Database B (Yale) [8,14] is a data set widely used for face recognition, and it consists of about 2414 frontal face images of 38 individuals under different illumination. We divide the data set into two parts, 50% of original data respectively, and perform 2-fold cross validation. The CMU Pose Illumination Expression (PIE) database [27] is a database of 41,368 images of 68 people. We only take a subset containing five near frontal poses (C05, C07, C09, C27, C29) under different illuminations and expressions and there are 170 images for each individual. We pick up 150 images per subject as training data. Digit recognition databases include the Optical Recognition of Handwritten Digits Data Set (Optdigits) [2], Pen-Based Recognition of Handwritten Digits Data Set (Pendigits) [2], and the USPS handwritten digit database [12]. We just separate the data into training set and testing set according to advice in [2,12]. Experiments are also performed on a more challenging data set Letter Recognition database [2] with 20,000 samples, in which we select 720 samples per letter as training set.

5.2. Classification accuracy

We compare our algorithms in three kinds of granularity, GLC-fine, GLC-coarse and GLC-intermediate (Parameter K in K-means is set as the number of classes here), with SRC [33], LRC [21] and CRC_RLS [40]. SRC are based on sparse representation, while LRC and CRC_RLS use similar framework with $\ell 2$-minimization, so it is necessary to make comparison among them to estimate efficacy of our methods. Algorithm are performed on features of PCA and LPP [11] in low dimension. Regularization parameter λ is set as 0.01.

Accuracy of different experiments are shown in Figs. 6–8. Generally, performance of GLC algorithms are similar, as is shown that those plots on figures nearly overlap, but most of the time GLC-intermediate has best performance. Also, GLC algorithms have comparable or sometimes better results than those of other methods.

As is shown in Fig. 6a, in experiments on Extended Yale Face Database B (YaleB for short on figures) with PCA feature, GLC algorithms give slightly better performance than that of SRC. In dimension of 30, accuracies of GLC-fine, GLC-coarse, GLC-intermediate are 81.05%, 82.37% and 81.97% respectively, while that of SRC is 80.35%. When it comes to LPP feature, in Fig. 6b, GLC in three granularity give the same performance, so the lines of them totally overlap. In dimension of 30, GLCs give accuracy of 96.1%, while that of SRC is 95.61% and those of LRC and CRC_RLS are 95.57% and 95.21% respectively. The difference becomes larger when dimension decreases. Plots of SRC and CRC_RLS overlap together. For PIE database of PCA feature in Fig. 6c, accuracies of GLC algorithms are similar with that of SRC, but still higher than those of LRC and CRC_RLS. But with LPP feature, the improvement is non-trivial; in dimension of 30, accuracies of GLC-fine and GLC-coarse are 95.27%, and that of GLC-intermediate is 95.35%, subtly higher, while accuracy of SRC is 92.91% and those of LRC and CRC_RLS are lower.

In digit recognition data sets, recognition rates on PCA feature of GLCs and SRC are close, as is shown in Fig. 7a and e. In the dimension of 30, GLC-intermediate gives accuracy of 97.77% on Optdigits, the same as that of SRC. While on USPS, accuracy of GLC-intermediate is 95.37%, only 0.25% higher than those of SRC. On LPP feature, GLCs give comparatively better performance. In the dimension of 30, GLC-intermediate achieve accuracies of 96.83% and 90.18% on Optdigits and USPS, 1.11% and 3.29% higher than those of SRC. Original dimension of data in Pendigits is 16, so experiments are conducted in dimension of 4, 8 and 12 with PCA and LPP feature and on the original data. For symmetry the accuracies on original data are duplicate in Fig. 7c and d. GLC-intermediate gives accuracies of 97.83%, 0.94% better than that of SRC on original data. Difference of accuracy is enlarged as the dimension goes lower.

Experiments on Letter Recognition data set are also conducted on original data and on feature of PCA and LPP in dimension of 4, 8 and 12, as is shown in Fig. 8. In dimension of 12, GLC-intermediate gets accuracies of 84.69% and 78.98% on the

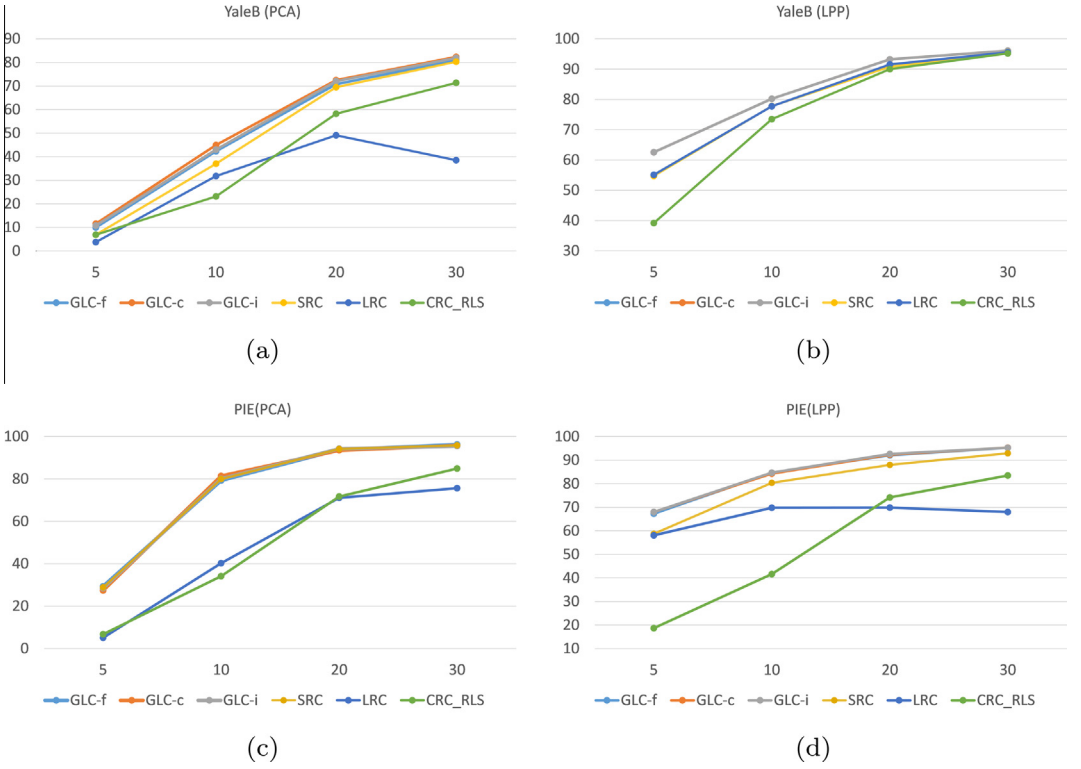

Fig. 6. Accuracy (%) on face recognition datasets. *X*-coordinate is the increasing dimension of feature. *Y*-coordinate is the accuracy.

two feature, 1.88% and 15.94% better than those of SRC. On other dimension, GLCs also show more efficacy than other algorithms do.

On the whole, GLC algorithms of different granularity show competitive performance with other methods.

5.3. Closeness of coefficients

In this section, we want to show that GLC algorithms actually assemble more nonzero elements to the neighbourhood of the target. A quantified measurement is used here.

Definition 2 (*Closeness Index (CI)*). The Closeness Index of reconstruction coefficient $\mu \in R^n$ is defined as

$$CI(\mu) = \frac{\sum_{j=1}^{n} |\mu_j| * d_j}{\|\mu\|_1} \tag{10}$$

The Closeness Index is exactly the weighted average of distances based on distribution of absolute value in μ, and it mainly reflects closeness of coefficients. Closeness Index also shows proper sparsity resulting from closeness, since μ will be also sparse if a target is only represented by its neighbourhood, which is of small amount. Such measure will be smaller, if more nonzero elements, or elements with large absolute value, assemble around the nearby samples. Conversely, if the distribution of nonzero elements is not characterized by both locality and certain sparsity, namely, either that the coefficients are dense or that they are sparse but nonzero coefficients of them spread on remote samples, the CI measure of such vector is still of large value. Generally speaking, the smaller CI measure is, the more coefficients are locality-preserving, and vice versa.

CI measure is not affected by magnitude of elements, because it is regularized by the $\ell 1$-norm of the vector. For precision, we normalize each data entry to have $\ell 2$-norm. Closeness Index of coefficients in experiments on different datasets are shown in Fig. 9. For Pendigits and Letter Recognition dataset, we draw the coefficients in experiment on the original data whose dimension is 16. PCA and LPP feature in dimension of 30 are shown for other datasets.

Generally, coefficients yielded by GLC-fine always give the lowest measurement, followed by those of GLC-intermediate. CI measures of SRC are larger than those of GLC-fine and GLC-intermediate. Although SRC gives sparse representation, it does not ensure closeness explicitly, part of nonzero elements it assigns may scatter on samples far away. In some cases, CI measurements of GLC-coarse are a little higher than those of SRC, which may result from the class-wise assignment of α. The class-wise assignment of α may cause that samples of the same label are treated equally as the closest one, regardless of

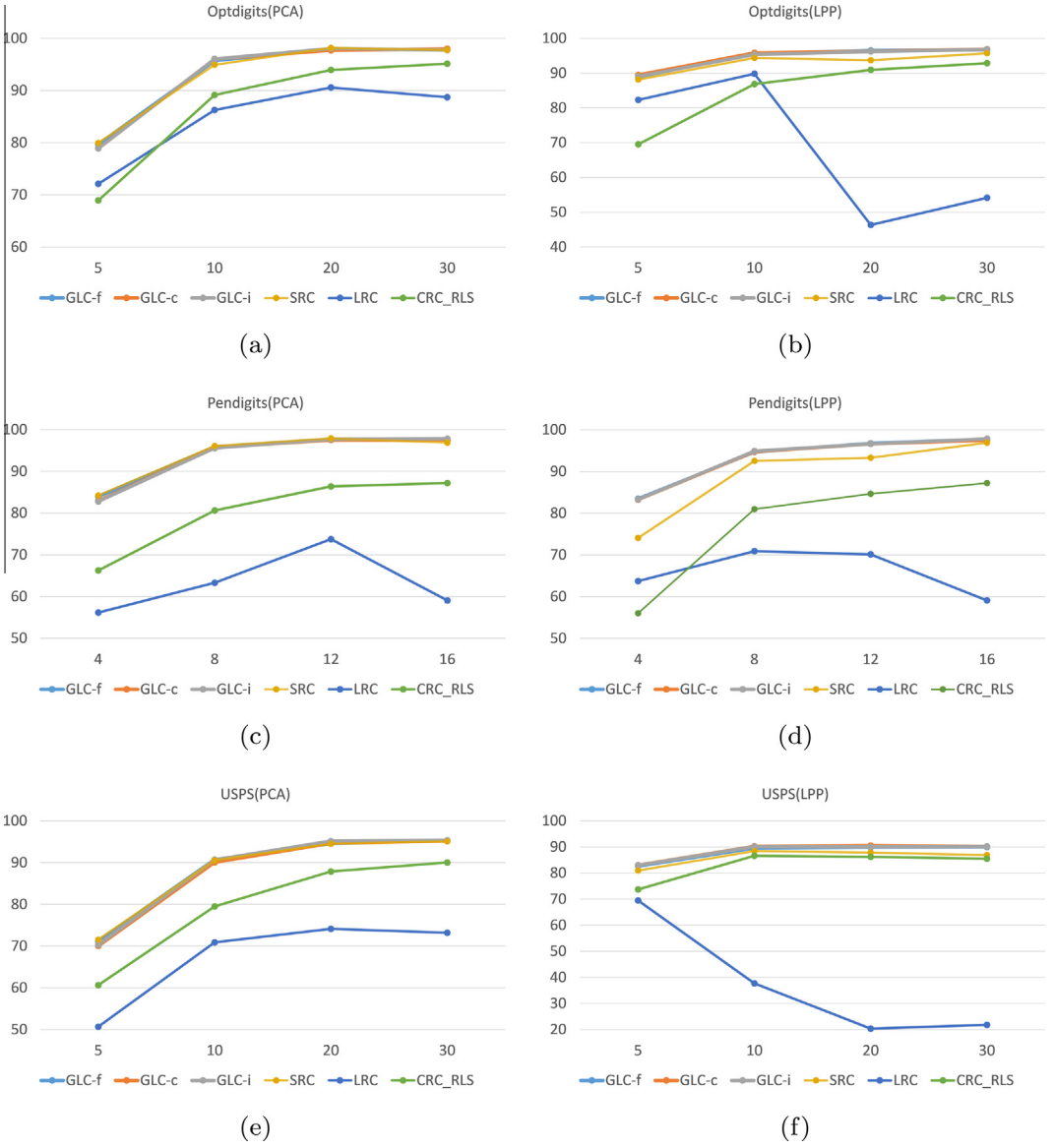

Fig. 7. Accuracy (%) on digit recognition datasets. *X*-coordinate is the increasing dimension of feature, and *Y*-coordinate is the accuracy.

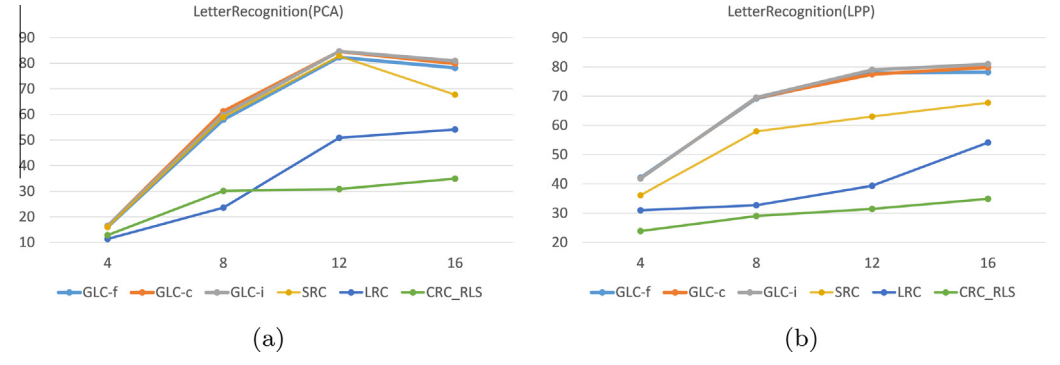

Fig. 8. Accuracy (%) on Letter Recognition dataset. *X*-coordinate is the increasing dimension of feature, and *Y*-coordinate is the accuracy.

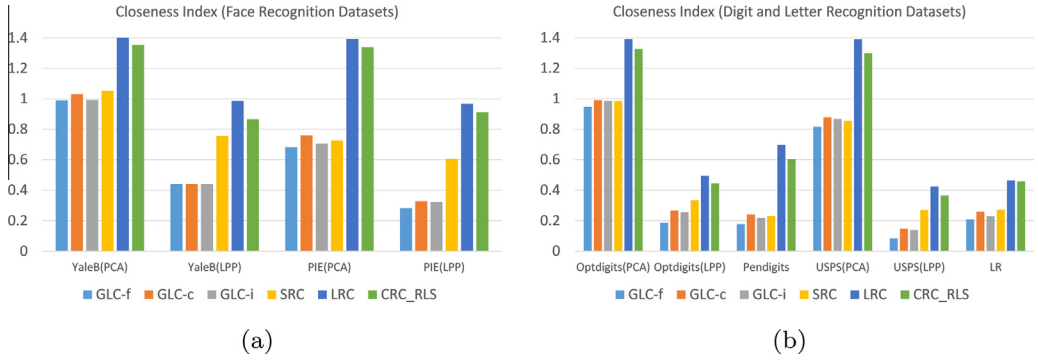

Fig. 9. Closeness Index of coefficients yielded by every dataset. LR is short for Letter Recognition database. *Y*-coordinate is the value of CI measurement.

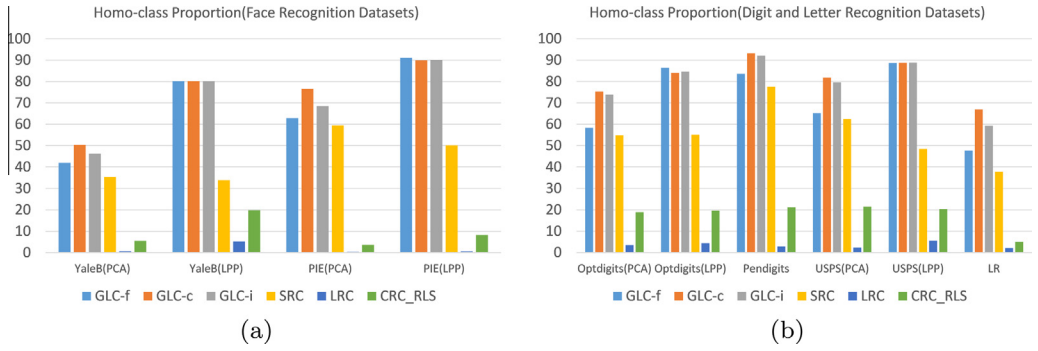

Fig. 10. Homo-class Proportion (%) of coefficients yielded by every dataset. LR is short for Letter Recognition database. *Y*-coordinate is the value of Homo-class Proportion.

individual distance, and remote samples may also get high values as the nearby ones do. LRC and CRC_RLS attain much high CI measures than previous algorithms do. These two methods adopt $\ell 2$-minimization to give dense representation, so their nonzero elements spread on many samples, and naturally show less closeness. What is more, coefficients from LPP feature totally show more closeness than from PCA feature, simply because LPP preserves locality in its projection while PCA does not.

5.4. Homo-class proportion

In this part, we examine the distribution of nonzero elements by evaluating the Homo-class Proportion, which is the proportion $\ell 1$-norm of elements concentrating on homo-class samples. Also, we draw the coefficients in experiments on the original data of Pendigits and Letter Recognition dataset while PCA and LPP feature in 30 dimension of other datasets.

As we can see in Fig. 10, GLC algorithms significantly adjust the distribution of nonzero elements and enable nonzero elements to gather more around the homo-class samples in all tests. Proportions of LRC and CRC_RLS are the lowest because of their dense representation. GLC-coarse enjoys the highest Homo-class Proportion on most of experiments, followed by GLC-intermediate, GLC-fine and SRC.

6. Discussion

6.1. Other assignment of α except Euclidean distance

We choose the Euclidean distance ($\ell 2$-distance) as assignment of α at default in GLC algorithms, but of course, there are many other kinds of distance as candidates, some of which will be compared briefly here.

For GLC-fine, we also can take Manhattan distance ($\ell 1$ distance) or the extension Minkowski distance as α. What is more, if the data is believed to lie on manifolds in the feature space, the approximated geodesic distance [29] can be utilized to estimate the similarity among samples. These distance can also be applied to GLC-coarse and GLC-intermediate when they takes the minimum distance of each group as assignment. On other hand, in GLC-coarse and GLC-intermediate we take the smallest distance among samples at default, but the distances from target to the centroid of groups are also available. One can also use the distances between the target and its projection on the subspace spanned by the groups of samples. Such idea

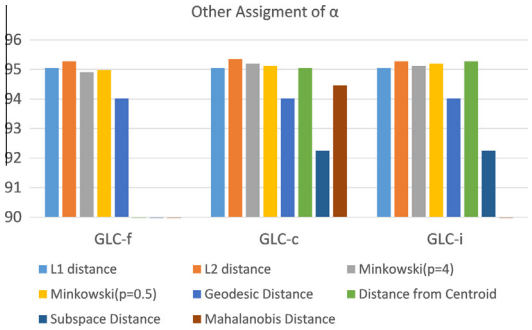

Fig. 11. Accuracy of different assignment of α with three granularity of GLC.

is similar with the Nearest Subspace. The difference is that Nearest Subspace directly classifies the target by examining the minimal subspace distance, while ours take such distances to adjust the distribution of nonzero elements. What is more, taking into account of the covariance of data, Mahalanobis distance can also be the assignment. Different assignment of α is based on different hypothesis and motivation.

We give a brief comparison among the assignment mentioned above and the Euclidean distance on PIE with LPP feature in dimension of 30. Accuracy is shown in Fig. 11. When using Mahalanobis distance the accuracy of GLC-intermediate is very low, so it is not shown here. Generally, accuracies among different assignments are similar, but the $\ell 2$ distance gives the best performance here. In GLC-intermediate, since the granularity is neither too coarse nor too fine, the distance from centroid of group is similar with the minimum distance, and these two distance give the same performance here. Still, we believe Euclidean distance is one of the proper assignments of α.

6.2. Selection of K in intermediate granularity

GLC-intermediate is required to choose parameter K in K-means step. It reduces to GLC-coarse when K is one and to GLC-fine when K is number of samples. To certain extent, K controls the granularity of division and we recommend that K is set as the number of class at default to gain partition of intermediate granularity. Here two simple comparisons are presented to show the effect of various K on accuracy of classification.

In Fig. 12, we show plots of results of GLC-intermediate on PIE and Optdigits data set respectively, varying K from different scale. For PIE, magnitude of K varies from 1000 to 10,000 and also includes 1, 68 (number of labels), 136 (two times of number of labels) and 10,200 (number of training samples). In Fig. 12a, accuracy of GLC-intermediate fluctuates a little with different K, and the size of groups decrease naturally. The best performance 95.35% is achieved when $K = 68$, the standard GLC-intermediate. For Optdigits, we vary K from 100 to 3800 and also include 1, 10 (number of labels), 20 (two times of number of labels) and 3823 (number of samples). The accuracy also waves slightly while size of groups goes down according to K as shown in Fig. 12b. The best performance is 97.11% when K is 500 or 1500. The selection of K controls the granularity of division and to some extent affects the accuracy of algorithm. To estimate the optimal K is a challenging problem since such problem turns out to be choosing the optimum granularity and proper segment of training samples.

From another aspect, except K-means, other unsupervised learning approach can be utilized to divide the data into partition of intermediate granularity. Nevertheless, the purpose of division in intermediate granularity is to retain certain locality in each group and such division is hoped to be simple, fast and efficient. As a result, K-means is a good choice here.

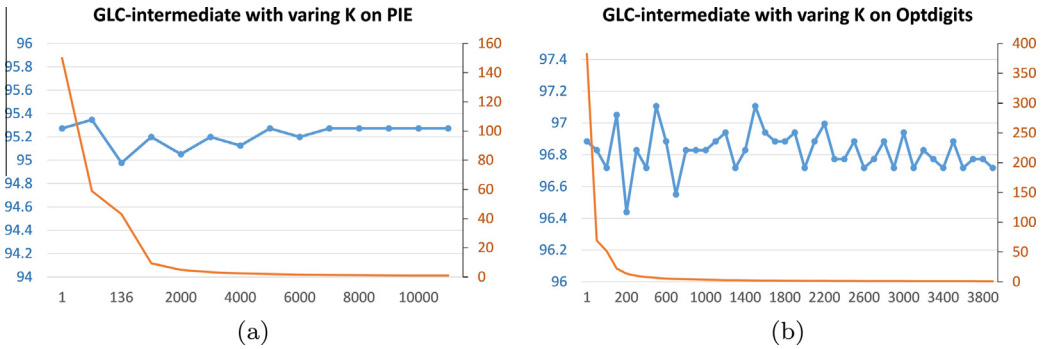

Fig. 12. (a) Accuracy (%) of GLC-intermediate with varying K on PIE. (b) Accuracy (%) of GLC-intermediate with varying K on Optdigits. X-coordinate is K in K-means step of GLC-intermediate. Y-coordinate is the accuracy (%) on left and the average size of groups on right.

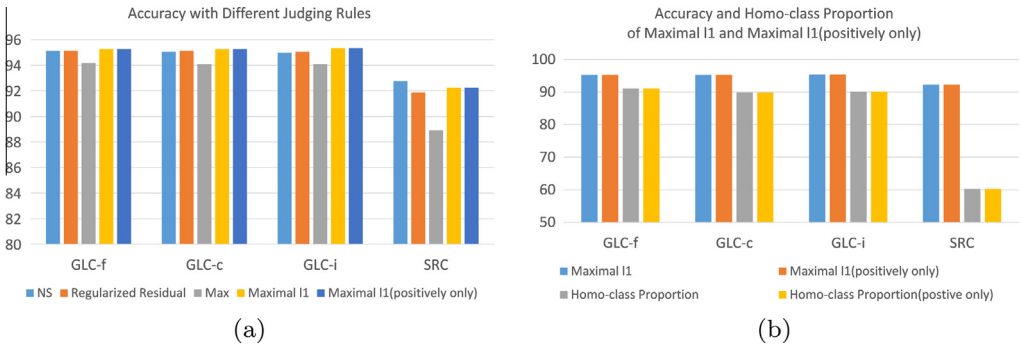

Fig. 13. (a) Accuracy (%) with different judging rules. *Y*-coordinate refers to accuracy. (b) Accuracy (%) and Homo-class Proportion (%) of maximal $\ell 1$ and maximal $\ell 1$ (positively only).

6.3. Comparison among judging rules

With coefficients, GLC algorithms classify the target as the label which has the maximum $\ell 1$-norm of coefficients. But other rules are also available, and we want to give a brief comparison here. Other than maximal $\ell 1$, GLC algorithms can still take the widely-used Nearest Subspace rule. Another simple idea is to choose the maximal element and take the corresponding label. This is similar with the Nearest Neighbour approach since GLC algorithms rely on the Closeness Assumption. Regularized residual taken by CRC_RLS [40] is a novel approach, which regularizes the residual with $\ell 2$ norm of coefficients.

In Fig. 13a, we show the accuracy of GLC algorithms, SRC with above rules in experiments on PIE database, feature of which is given by LPP in dimension of 30. The Homo-Class Proportion is also provided for comparison. In GLC algorithms, maximal $\ell 1$ shows slightly better accuracy than Nearest Subspace. The difference is 0.15% in GLC-fine, 0.22% in GLC-coarse and 0.37% in GLC-intermediate. Even attached with SRC, maximal $\ell 1$ can also give comparable accuracy with that of Nearest Subspace, suggesting that such rule may be generalized as an alternative to Nearest Subspace.

Furthermore, in Fig. 13b, it is surprising to see that, to check the $\ell 1$-norm of only positive elements of each labels, i.e. to simply sum up the positive elements respectively and select the largest one, can give similar performance with maximal $\ell 1$. Such phenomenon may be explained by the fact that the Homo-class Proportions of positive elements are also very high, suggesting that distribution of positive elements is also skewed towards homo-class samples. Similar phenomenon is also reported in [38] that neighbourhood coefficients tend to be nonnegative. The principle behind is worth further study. Generally, no matter what kind of rule is adopted, higher Homo-class Proportion, which means nearly homo-class representation of each target, tends to result in better accuracy of classification.

7. Conclusion

In this paper, we present influence of distribution of nonzero elements on classification accuracy and our solution to adjust this distribution. The proposed Granular Locality-preserving Classification (GLC) algorithm incorporates distance metric, class labels and clustering results of *K*-means on training data as discriminative information into the objective function, and judges the target as the class having maximal $\ell 1$-norm. Such method preserves locality with automatic adjustment methods. GLC algorithms of different granularity show comparable efficacy with state-of-the-art algorithms in experiments on different classification problems, and they alter the distribution of nonzero elements as expected. In future work, we will explore other methods to further adjust the distribution, such as to impose isotonic constraint which ensures that the nonzero coefficients decrease according to distances from the target.

Acknowledgments

This work is funded by the National Basic Research Program of China (973 Program) under Grant No. 2012CB316400, and the National Natural Science Foundation of China (NSFC) under Grant No. 61375052.

References

[1] Edoardo Amaldi, Viggo Kann, On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems, Theor. Comput. Sci. 209 (1) (1998) 237–260.
[2] K. Bache, M. Lichman, UCI Machine Learning Repository, 2013.
[3] Douglas Brown, Locality-Regularized Linear Regression for Face Recognition, ICPR, 2012, pp. 1586–1589.
[4] Weihong Deng, Jiani Hu, Jun Guo, Extended SRC: Undersampled face recognition via intraclass variant dictionary, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2012) 1864–1870.
[5] Weihong Deng, Jiani Hu, Jun Guo, In defense of sparsity based face recognition, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2013, pp. 399–406.

[6] David L. Donoho, For most large underdetermined systems of linear equations the minimal ℓ1-norm solution is also the sparsest solution, Commun. Pure Appl. Math. 59 (6) (2006) 797–829.
[7] Shenghua Gao, Ivor Waihung Tsang, Liang-Tien Chia, Peilin Zhao, Local features are not lonely–Laplacian sparse coding for image classification, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 3555–3561.
[8] Athinodoros S. Georghiades, Peter N. Belhumeur, David J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[9] Michael Grant, Stephen Boyd, Graph implementations for nonsmooth convex programs, in: V. Blondel, S. Boyd, H. Kimura (Eds.), Recent Advances in Learning and Control, Lecture Notes in Control and Information Sciences, Springer-Verlag Limited, 2008, pp. 95–110.
[10] Michael Grant, Stephen Boyd, CVX: Matlab Software for Disciplined Convex Programming, Version 2.1, March 2014. <http://cvxr.com/cvx>.
[11] Xiaofei He, Partha Niyogi, Locality preserving projections, in: NIPS, vol. 16, 2003, pp. 234–241.
[12] J.J. Hull, A database for handwritten text recognition research, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 550–554.
[13] Seung-jean Kim, K. Koh, M. Lustig, Stephen Boyd, Dimitry Gorinevsky, An interior-point method for large-scale ℓ1-regularized least squares, IEEE J. Sel. Top. Signal Process. 1 (4) (2007) 606–617.
[14] Kuang-Chih Lee, Jeffrey Ho, David Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005) 684–698.
[15] S.Z. Li, Face recognition based on nearest linear combinations, Comput. Vision Pattern Recogn. (1998) 839–844.
[16] Huaping Liu, Yulong Liu, Fuchun Sun, Traffic sign recognition using group sparse coding, Inform. Sci. 266 (2014) 75–89.
[17] Yanan Liu, Fei Wu, Zhihua Zhang, Yueting Zhuang, Shuicheng Yan, Sparse representation using nonnegative curds and whey, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 3578–3585.
[18] Long Ma, Chunheng Wang, Baihua Xiao, Wen Zhou, Sparse representation for face recognition based on discriminative low-rank dictionary learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2586–2593.
[19] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, Online learning for matrix factorization and sparse coding, J. Mach. Learn. Res. 11 (2010) 19–60.
[20] Xue Mei, Haibin Ling, Robust visual tracking and vehicle classification via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (11) (2011) 2259–2272.
[21] Imran Naseem, Roberto Togneri, Mohammed Bennamoun, Linear regression for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (11) (2010) 2106–2112.
[22] Bruno A. Olshausen, David J. Field, Sparse coding of sensory inputs, Curr. Opin. Neurobiol. 14 (4) (2004) 481–487.
[23] Armin Saeb, Farbod Razzazi, Massoud Babaie-Zadeh, SR-NBS: a fast sparse representation based n-best class selector for robust phoneme classification, Eng. Appl. Artif. Intell. 28 (2014) 155–164.
[24] D. Seung, L. Lee, Algorithms for non-negative matrix factorization, Adv. Neural Inform. Process. Syst. 13 (2001) 556–562.
[25] Soheil Shafiee, Farhad Kamangar, The role of dictionary learning on sparse representation-based classification, in: Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments, 2013.
[26] Qinfeng Shi, Anders Eriksson, Anton van den Hengel, Chunhua Shen, Is face recognition really a compressive sensing problem?, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2011, pp 553–560.
[27] Terence Sim, Simon Baker, Maan Bsat, The CMU Pose, Illumination, and Expression (PIE) Database, 2002.
[28] Josef Stoer, Roland Bulirsch, R. Bartels, Walter Gautschi, Christoph Witzgall, Introduction to Numerical Analysis, vol. 2, Springer, New York, 1993.
[29] Joshua B. Tenenbaum, Vin De Silva, John C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[30] William E. Vinje, Jack L. Gallant, Sparse coding and decorrelation in primary visual cortex during natural vision, Science 287 (5456) (2000) 1273–1276.
[31] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, Yihong Gong, Locality-constrained linear coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 3360–3367.
[32] John Wright, Yi Ma, Julien Mairal, Guillermo Sapiro, Thomas S. Huang, Shuicheng Yan, Sparse representation for computer vision and pattern recognition, Proc. IEEE 98 (6) (2010) 1031–1044.
[33] John Wright, Allen Y. Yang, Arvind Ganesh, Shankar S. Sastry, Yi Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[34] Yong Xu, Qi Zhu, Zizhu Fan, David Zhang, Jianxun Mi, Zhihui Lai, Using the idea of the sparse representation to perform coarse-to-fine face recognition, Inform. Sci. 238 (2013) 138–148.
[35] Jian Yang, Lei Zhang, Yong Xu, Jing-yu Yang, Beyond sparsity: the role of L1-optimizer in pattern classification, Pattern Recogn. 45 (3) (2012) 1104–1118.
[36] Meng Yang, Lei Zhang, Jian Yang, David Zhang, Robust sparse coding for face recognition, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2011, pp. 625–632.
[37] Shuyuan Yang, Yuan Lv, Yu Ren, Lixia Yang, Licheng Jiao, Unsupervised images segmentation via incremental dictionary learning based sparse representation, Inform. Sci. 269 (2014) 48–59.
[38] Kai Yu, Tong Zhang, Yihong Gong, Nonlinear Learning using Local Coordinate Coding, in: NIPS, vol. 9, 2009, pp. 1.
[39] Bob Zhang, Fakhri Karray, Qin Li, Lei Zhang, Sparse representation classifier for microaneurysm detection and retinal blood vessel extraction, Inform. Sci. 200 (2012) 78–90.
[40] Lei Zhang, Meng Yang, Xiangchu Feng, Sparse representation or collaborative representation: which helps face recognition?, in: IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp 471–478.
[41] Luming Zhang, Mingli Song, Xiao Liu, Li Sun, Chun Chen, Jiajun Bu, Recognizing architecture styles by hierarchical sparse coding of blocklets, Inform. Sci. 254 (2014) 141–154.
[42] Qi Zhu, Yong Xu, Jinghua Wang, Zizhu Fan, Kernel based sparse representation for face recognition, in: 2012 21st International Conference on Pattern Recognition (ICPR), IEEE, 2012, pp. 1703–1706.
[43] Hui Zou, Trevor Hastie, Robert Tibshirani, Sparse principal component analysis, J. Comput. Graph. Stat. 15 (2) (2006) 265–286.